

Decision-making (for Making of AI Programming)

Decision-making is about deciding the order of execution of statements based on certain conditions. PYTHON language handles decision-making by supporting the following statements.

1. **If** statement
2. **If--- else** statement
3. **if--- elif--- else** statement
4. **if--- else conditional Expression** statement

When you write a computer program, regardless of the programming language, you often need to execute different set of statements depending on some satisfied condition. The process of determining the order in which statements execute in a program is called decision-making or flow of Control. The most common type of decision statement is the **If** statement which you study following Chapter.

❖ Decision making with **If** statement

1. **If** statements

The **If** statement may be implemented in different forms depending on the evaluation of a conditional expression return a value of **True**. The syntax for a simple **If** statement is as follows:

Syntax:- *For one statement without block.*

If conditional expression:

----- Statement

Syntax:- *For block or more than one statement.*

If conditional expression:

----- Statement1

----- Statement2

-----:

Statement-3 # statement-3 out of **If** statement

Flowchart of **If statement**

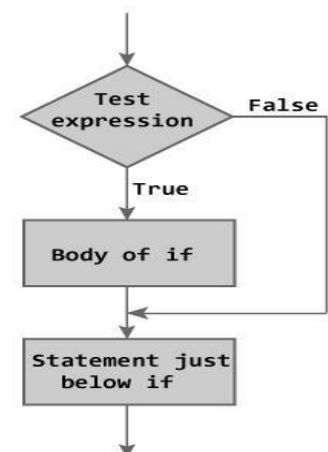


Figure: Flowchart of if Statement

If the expression is evaluated and found to be true, the single statement following the "if" is executed. If false, the following statement is skipped. Here a compound statement composed of several statements bounded by indentation statement.

©Copy Right

Example:- *If* condition is True than execute for one statements.

```
Number = 2    ##//Number stored 2 "number"
If Number>0:    //Checks whether number you entered is greater than
----- print ("Number, This number is Positive ")
```

Example:- *If* condition is true than execute for block or more than one statements.

```
Roll=2
If Roll==2:
    ----- print (" Name is : Nadeem Mustafa ")
    ----- print (" Father Name is : Fahim Mustafa")
    ----- print (" Address : F.B.Area Karachi ")
```

2. *if---else* statement:-

This feature permits the programmer to write a single comparison, and then execute one of the two statements depending upon whether the test expression is true or false. The general form of the *if---else* statement is.

Syntax:-

***If* Test-Expression:**

----- statement1

else:

-----statement2

Flowchart of *if...else* statement

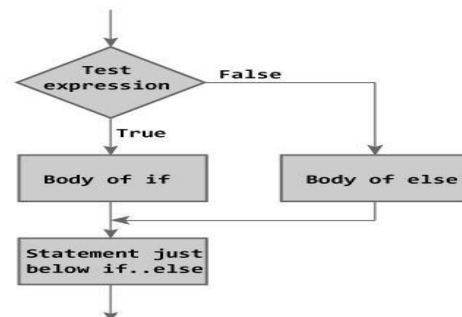


Figure: Flowchart of *if...else* Statement

Here also expression in parentheses must evaluate to (a Boolean) true or false. Typically you are testing something to see *If* it's true, and then running a code block (one or more statements) *If* it is true, and another block of code *If* it isn't. The statement1 or statement2 can be either simple or compound statement.

The following program demonstrates a legal *If else* statement:

Example-1

```

number=0
if number == 0:
    ----print ("Given number is Zero ")
else:
    ----print ("Given number is not zero ")

```

```

//Number stored 0 "number"
//Checks whether number you entered is equal to 0 or Not.
// If YES then enters into if-body
//If not then enters into else -body

```

Example-2

```

day = "Sunday"
if day == "Saturday" or day == "Sunday":
    print("It's a weekend")
else:
    print("It's a working day")
Output:
It's a weekend

```

Explanation: The second condition is true because day == "Sunday".

3. If --- elif--- else:

This brings up the other **if-else** construct; **if---elif---elif--- else** This construct is useful where two or more alternatives are available for selection.

Syntax: If condition1: ----statement1 elif condition2: ----statement2: elif condition3: ----statement 3 else: ----statements4	Example Letter="A" If Letter=="A": ----print("A for Allah") If Letter=="B": ----print("B for Bismillah") If Letter=="C": ----print("C for Captain") else: ----print("Unknown Letter")
--	--

The various conditions are evaluated one by one starting from top to bottom, on reaching a condition evaluating to true the statement group associated with it are executed and skip other statements. If none of expression is evaluate to true, then the statement or group of statement associated with the final **else** is executed.

Difference between **If...** and **elif...else** is that :

If statement example	If elif....else statement example
<pre> per = 72 if per >=80: print ("Grade A+ ") if per >=70: print ("Grade A ") if per >=60: print ("Grade B ") if per >=50: print ("Grade C ") Output: Grade A Grade B Grade C // it will checks all conditions and execute all conditions that are true// </pre>	<pre> per = 72 if per >=80: print ("Grade A+ ") else if per >=70: print ("Grade A ") else if per >=60: print ("Grade B ") else if per >=50: print ("Grade C ") else print("Fail") Output: Grade A // It stop when the condition is true (i.e. not check further) and then exit from the If – structure </pre>

Nested If Expression :

If there is another structure within **If** structure that is called nested **If** statement.

Syntax:

```

If expression :           ← Outer If
    if expression1 :      Inner If ←
        statement block1
    else:
        statement block2

← Outer If ends here
else:
    statement block3;

```

If 'expression' is false the 'statement-block3' will be executed, otherwise it continues to perform the test for 'expression 1' . **If** the 'expression 1' is **true** the 'statement-block1' is executed otherwise 'statement-block2' is executed.

Example

The **if-else** statement can also use to test for Nested conditions. The following example uses two conditions so that **if** the first test fails, we want to perform a second test before deciding what to do:

```
n=91
if %2 != 0:
    if n>10:
        print (n, " is an odd number and greater than 10 ")
    else:
        print (n," is an odd number and less than 10 ")
else:
    print (n," n is an even number")
```

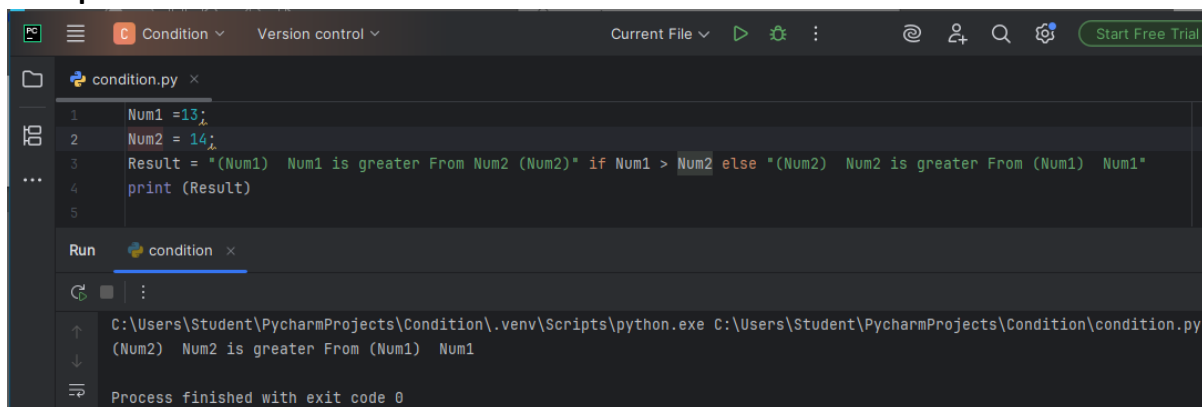
4. Conditional Expression (Short-hand if-else)

We have covered **conditional expression** in the previous condition expression which can be used to replace **if...else** statements. A conditional expression lets you choose a value using a single line of code.

Syntax:

value_if_true = if condition else value_if_false

Example:



The screenshot shows a code editor with a file named 'condition.py'. The code in the file is:

```
1 Num1 =13;
2 Num2 = 14;
3 Result = "(Num1) Num1 is greater From Num2 (Num2)" if Num1 > Num2 else "(Num2) Num2 is greater From (Num1) Num1"
4 print (Result)
5
```

Below the code editor, the 'Run' output is displayed:

```
Run condition
C:\Users\Student\PycharmProjects\Condition\.venv\Scripts\python.exe C:\Users\Student\PycharmProjects\Condition\condition.py
(Num2) Num2 is greater From (Num1) Num1
Process finished with exit code 0
```

It will check **if** (Num1> Num2) then it will evaluate *Expression 1* i.e. , but since Num1 is not greater than from Num2 here so it will evaluate *expression 2* i.e.

Points to Remember

©Copy Right

✓ It is also called “**Ternary Expression statement**” as it works on three operands.

Membership Conditions (in and not in)

Membership operators check whether a value exists in a sequence, such as a list or string.

Example using *in*:

```
fruits = ["apple", "banana", "mango"]  
if "banana" in fruits:  
    print("Banana is available")
```

Output:

Banana is available

Example using *not in*:

```
name = "Ali"  
if "z" not in name:  
    print("The letter z is not present")
```

Output:

text

The letter z is not present

Final tip: Start by mastering comparison operators (`==`, `!=`, `>`, `<`, `>=`, `<=`), then learn **and**, **or**, and **not**. These are the foundation for writing conditions/If in Python.

Exercise

Theory Questions

1. Decision-making structures cannot be nested. True or false with give any example.
2. How do you perform more than one statement when a condition is true?
3. Correct the syntax error line by line.
 - 1) If (x > 25)
 - 2) y = x;
 - 3) else
 - 4) y= z.
4. What is difference between the if – else statement and if-- elif--else

Practical Questions

- 1) Write a simple program to construct a **Calculator** that can perform Basic Mathematical operation as well.
- 2) Write a simple program to check an alphabet entered by user is a **Vowel** or not using **If**
- 3) Write a code to check an integer number entered by user is **Even** or **Odd** using **Conditional expression statement**.
- 4) Write a program to input subject marks Math, English and Physics then to prepare marks sheet of a student that will show the Obtain marks, percentage and grade.
- 5) Write a program that input a number, then report whether the number is in the range from 1 to 100. Otherwise the number above to 100.
- 6) 6. Write a if – elif—else statement that output messages indicating what day has been numerically input for example (1-> Monday, 2-> Tuesday, 3-> Wednesday,...)

Objective MCQ's

1. In a simple **If** statement with no **else**, what happens **If** the condition following the **If** is false?
 - a) The program search for the last **else** in the program.
 - b) Nothing, or control falls through to the statements following the if.
 - c) The body of the **If** statement is executed.
 - d) The program as a whole is executed.

2. The conditional operator statement that

```
x=0 ;  
result = "x equal to zero" if x==10: else "x not equal to zero" ;  
print (result)
```

- a) Is incorrect syntax
- b) Is correct syntax, but x not equal to zero will be print
- c) Cause a run time error
- d) Has no effect on the program

3. What value is assign in income_tax to salary by the **If** statement when salary is 55000?

```
if salary > 70000 :  
    income_tax = 1000  
elif salary > 50000 :  
    income_tax = 500  
else  
    income_tax = 0
```

- a) 1000
- b) 500
- c) 0
- d) Nothing above all

4. Which is type of ternary Expression

- a) Logical operator
- b) Assignments operator
- c) Relational
- d) Conditional expression

5. What is the output?

```
X=10  
if x>5:  
    print("Yes")  
else:  
    print("No")
```

- a) Yes
- b) No
- c) True
- d) Nothing all